

(19)



Europäisches Patentamt
European Patent Office
Office européen des brevets



(11) Veröffentlichungsnummer: **0 645 702 A1**

(12)

EUROPÄISCHE PATENTANMELDUNG

(21) Anmeldenummer: 93115467.8

(51) Int. Cl.⁶: **G06F 9/46, H04M 3/36,
H04Q 3/545**

(22) Anmeldetag: 24.09.93

(43) Veröffentlichungstag der Anmeldung:
29.03.95 Patentblatt 95/13

(84) Benannte Vertragsstaaten:
AT BE CH DE DK ES FR GB IT LI NL SE

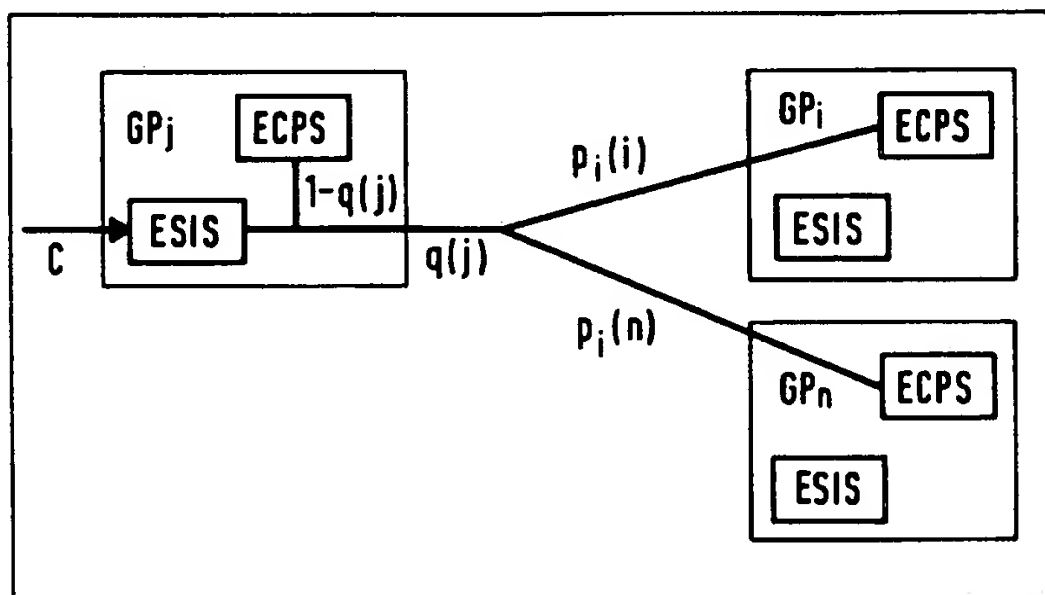
(71) Anmelder: **SIEMENS AKTIENGESELLSCHAFT**
Wittelsbacherplatz 2
D-80333 München (DE)

(72) Erfinder: **Witte, Martin, Dr.**
Josephsburgstrasse 5
D-81673 München (DE)
Erfinder: **Oehlerich, Jörg**
Präntlweg 1
D-82131 Stockdorf (DE)
Erfinder: **Held, Walter**
Isardamm 129
D-82538 Geretsried (DE)

(54) Verfahren zum Lastausgleich in einem Multiprozessorsystem.

(57) Verfahren zum Lastausgleich in einem Multiprozessorsystem, insbesondere ein Multiprozessorsystem eines Kommunikationssystems, bei dem anfallende Aufgaben von mehreren Prozessoren unter Realzeitbedingungen abgearbeitet werden können. Geraten aufgrund von lokalen Starklastsituationen einzelne Prozessoren in Überlast, so wird ein Abbau der lokalen Überlast dadurch erreicht, daß in dieser Lastsituation weitere eintreffende Anforderungen an den Prozessor so weit als möglich/nötig an andere, nicht überlastete Prozessoren weitergeleitet werden.

FIG 3



EP 0 645 702 A1

Die Erfindung betrifft ein Verfahren zum Lastausgleich in einem Multiprozessorsystem, bei dem anfallende Aufgaben von mehreren Prozessoren unter Realzeitbedingungen abgearbeitet werden können.

Wenn es sich bei dem Multiprozessorsystem um ein prozessorgesteuertes Kommunikationssystem handelt, wird das Kommunikationssystem aufgrund eines prognostizierten Vermittlungsaufkommens und/oder einzuhaltender Mindestanforderungen, wie beispielsweise die an die Verbindungsaufbauzeiten (Through Connection Delay), dimensioniert. Ein wesentlicher Parameter der die Dimensionierung eines Vermittlungssystems bestimmt, wird mit dem zu erwartenden Verkehrswert vorgegeben. Dieser Verkehrswert ergibt sich dabei aus dem Quotienten der sich aus der Summe der Belegungsdauer - bezogen auf die Belegungen in einem Beobachtungsintervall errechnet. Ein nach den herkömmlichen Verkehrserwartungen dimensioniertes prozessorgesteuertes Kommunikationssystem muß jedoch weitgehendst auch alle ankommenden Vermittlungsanfragen bei auftretenden Spitzenbelastungen, die an bestimmten Tagen, einzelnen Tagesstunden oder nicht vorherbestimmbaren Ereignissen auftreten, beispielsweise unter Zuhilfenahme von Lastausgleichsverfahren, abarbeiten können.

Der Erfindung liegt die Aufgabe zugrunde, ein Lastausgleichsverfahren für ein Realzeitsystem anzugeben, das gewährleistet, daß auch in Starklastsituationen möglichst wenig Prozessoranforderungen abgewiesen werden.

Die Aufgabe wird durch die Merkmale des Patentanspruchs 1 gelöst.

Die Erfindung bringt den Vorteil mit sich, daß auf Grund der Berücksichtigung von Lastzuständen von Prozessoren die Überlast einzelner Prozessoren auf freie, nicht überlastete Prozessoren verteilt wird, wobei die verteilten Aufgaben keine zusätzlichen Lastschwankungen innerhalb von vermaschten Prozessoren verursachen sowie die Einhaltung der Through Connection Delays dadurch nicht beeinflußt werden.

Desweiteren bringt die Erfindung neben dem Vorteil, daß die in "Unterlast" befindlichen Prozessoren vor Überlastung durch verteilte Aufgaben geschützt werden, den weiteren Vorteil mit sich, daß die Auslastung des Prozessors an seiner "Hochlastschwelle" orientiert wird.

Eine Ausgestaltung der Erfindung ist, daß die Lastkennwerte mit einem Hysteresewert beaufschlagt werden, was den Vorteil mit sich bringt, daß Überschwingungen bei Eintritt in den Überlastfall vermieden werden.

Eine weitere Ausgestaltung der Erfindung ist, daß zusätzliche Aufgaben auch durch einen Prozessor abgearbeitet werden, dessen Lastzustand mit "high" eingestuft wird, dies bringt den Vorteil mit sich, daß die Anzahl der zugewiesenen fremden Aufgaben pro Prozessor an seiner zur Verfügung stehenden freien Kapazität orientiert werden.

Eine Ausgestaltung der Erfindung ist, daß der Anteil der von einem in Überlast befindlichen Prozessor wegzuverteilenden Aufgaben aus der Lastkennzahl ermittelt wird.

Eine Ausgestaltung der Erfindung ist, daß zur selbst bearbeiteten Last die verteilte Last geschätzt dazu addiert wird. Dies bringt den Vorteil mit sich, daß erst wenn die geschätzte tatsächlich am Prozessor anliegende Last unter eine Hochlastschwelle sinkt und eine vermittlungstechnische Überlasteinstufung auf Null gesetzt und eine Sicherheitsperiode eingehalten wurde, die Überlastsituation des Prozessors als beendet angesehen wird.

Vorteilhafte Weiterbildungen der Erfindung sind in den Unteransprüchen angegeben.

Weitere Besonderheiten der Erfindung werden aus der nachfolgenden näheren Erläuterung eines Ausführungsbeispiels, in Form eines Kommunikationssystems, anhand der Zeichnungen ersichtlich.

Es zeigen:

Fig 1. einen Teil eines Kommunikationssystems,

Fig 2. Prozeß-Verarbeitungseinheiten eines solchen zum Kommunikationssystem gehörenden Gruppenprozessors,

Fig 3. ein Schema für das erfindungsgemäße Verfahren, und

Fig 4. ein Ablaufdiagramm.

Fig 1 zeigt schematisch, in einem zum Verständnis der Erfindung erforderlichen Umfang Komponenten eines Kommunikationssystems K. Diese Komponenten sind Teilnehmerendgeräte $Tn1, \dots, Tnm$, Anschlußbaugruppen $LTG1, \dots, LTGn$ für den Anschluß der mit diesen Teilnehmerendgeräten verbindenden Teilnehmeranschlußleitungen, Gruppenprozessoren GPn , ein Koordinationsprozessor CP oder ein weiterer die Funktionen eines Koordinationsprozessors übernehmender Gruppenprozessor GP, eine Schnittstelleneinrichtung SE zu weiteren Vermittlungssystemen, eine Serviceeinheit SU, einem im Asynchronen Transfer Mode arbeitenden Multiplexer AMX sowie ein Koppelnetz SNB, daß mit einem Asynchronem Transfer Mode ATM Verbindungswege zu gewünschten Teilnehmerendgeräten Tnm durchschaltet. Die Anschlußbaugruppen $LTGn$ fassen jeweils eine Mehrzahl von Teilnehmerendgeräten zusammen. Vermittlungstechnische Aufgaben die in den Anschlußbaugruppen $LTGn$ anfallen werden jeweils durch mindestens einen Gruppenprozessor GPI bearbeitet. Die in einem Kommunikationssystem K verwendeten Gruppenprozessoren GPn sind

dabei im wesentlichen alle gleichartig strukturiert. Die Gruppenprozessoren GPn haben über den Multiplexer AMX Zugang zu weiterverarbeitenden Einheiten des Kommunikationssystems K. An diesen Multiplexer AMX sind neben den Gruppenprozessoren GPn die Service Einheit SU, das Koppelnetz SNB als auch weitere Prozessoren, wie beispielsweise ein Koordinationsprozessor CP angeschlossen. Über den Asynchronen Transfer Mode Multiplexer AMX besteht unter anderem auch ein Zugang zu anderen Vermittlungssystemen.

Die Gruppenprozessoren GPn sind hauptsächlich für den Verbindungsaufbau und Verbindungsabbau zuständig. Sie besitzen zudem zur Abwicklung solcher vermittlungstechnischer Aufgaben Kenndaten über die speziellen Ausprägungen der an ihnen angeschlossenen Anschlußbaugruppen LTGn. Der Datenaustausch zwischen den Gruppenprozessoren GPn findet über den AMX statt.

Fig. 2 zeigt einige für die Erfindung wesentliche Systemkomponenten des Gruppenprozessors GPi. Der Gruppenprozessor GPi wird intern in eine Vielzahl virtueller Prozesseinheiten VCPUs unterteilt, die jeweils für verschiedene Aufgaben zuständig sind. Die Zuweisung der Rechenzeit der "Central Prozessor Unit" CPU an die jeweiligen Mikroprozessoreinheit VCPUs erfolgt über ein spezielles Zuweisungsverfahren (Scheduling). Im Zusammenhang mit der Erfindung ist die virtuelle Prozesseinheit VCPU-CallP von besonderem Interesse, da sie für die Vermittlung der Telefonverbindungen zuständig ist. Die virtuelle Prozesseinheit VCPU-CallP setzt sich intern aus einer Vielzahl von Prozessen zusammen:

Beispielsweise seien hier einige Prozesse sowie deren Aufgaben aufgeführt:

- ESIS: Extended Signalling and Interworking Subsystem Das EWSX Signalling and Interworking Subsystem ESIS bildet die Schnittstelle zwischen den verschiedenen an die Anschlußbaugruppen LTGn zugeordneten Teilnehmerendgeräten Tn1,...,Tnm und führt die Anpassung zwischen externen Signalisierungssystemen und dem internen Message Interface durch.
- ECPS: Extended Call Processing Subsystem Dieses Subsystem ist für die Verarbeitung des Call-Routing, Call-Setups, Call-Cleardown sowie spezielle Features zuständig. Zur Abarbeitung dieser Prozesse können spezielle Dienstleistungen, wie die des ATS(Access Transaction Segment), UTS (User Transaction Segment), AS (Associator Segment)...., in Anspruch genommen werden.

Diese Prozesse kommunizieren miteinander über sogenannte Messages. Pro Verbindung eines Calls, einer von einem Teilnehmerendgerät Tnm ausgelösten Prozedur zum Verbindungsaufbau, werden dabei eine Vielzahl von Messages zwischen den einzelnen Prozessen verschickt. Dieser im Einzelfall aus weit über 100 Meldungsprozeduren bestehende Meldungsverkehr läuft pro Call auf einen einzigen Gruppenprozessor GPi ab. Wird ein Call verteilt, laufen die Messages zwischen dem Prozeß ESIS, der sich auf dem "Heim-Gruppenprozessor" GPj befindet und den ECPS Prozessen sowie weiteren vermittlungstechnischen Prozessen über das ATM Netz auf einen noch "belastbaren" Gruppenprozessor GPx ab.

Fig. 3 zeigt schematisch ein Verteilschema, daß an den Gruppenprozessor GPj anliegende Calls C an noch "belastbare" Gruppenprozessoren GPx weiterleitet.

Ein in den Gruppenprozessor GPn implementierter Load-Manager GPLM, der durch die virtuelle Prozesseinheit Operation & Maintenance (high priority) VCPU-OAMH jeweils eine Prozessorzeit zur Abarbeitung seiner Aufgaben zugewiesen bekommt, ermittelt den augenblicklichen Lastzustand (Gruppenprozessor Load State) GPLS des Gruppenprozessors sowie einen Wert OPL (Overload Priority Level) der eine spezifische vermittlungstechnische Überlastabwehrstufe angibt. Diese spezifische vermittlungstechnische Überlastabwehrstufe OPL wird jeweils im Lastzustand - "overload"- des Gruppenprozessors ermittelt. Folgende Lastzustände - normal, high, overload und extreme - des Gruppenprozessors GPn werden entsprechend der Belastung der zentralen Prozesseinheit CPU des Gruppenprozessors vom Load-Manager GPLM ermittelt. Die einzelnen Lastzustände GPLS des Gruppenprozessors GPx sind dabei beispielsweise wie folgt charakterisiert:

GPLS-normal:

Die Auslastung des Gruppenprozessors ist unterhalb seiner für ihn definierten Normallastschwelle. Die Normallastschwelle liegt beispielsweise bei 0,71 Erlang. (Erlang ist die Einheit des Verkehrswertes, wobei 0.71 Erlang einer Auslastung des Gruppenprozessors von 71 Prozent entspricht)

GPLS-high :

Der Gruppenprozessor befindet sich in Hochlast, das heißt oberhalb seiner Normallastschwelle. In diesem Zustand werden noch keine Überlastabwehrmaßnahmen für den Gruppenprozessor eingeleitet.

GPLS-overload:

Der Gruppenprozessor befindet sich in Überlast. Aufgaben die vom Gruppenprozessor zu erledigen wären, werden soweit möglich, kurzfristig auf andere, nicht überlastete Gruppenprozessoren GPn übertragen. Die Überlastschwelle liegt beispielsweise bei $> 0,8$ Erlang.

Der auf dem Gruppenprozessor GPn implementierte Load Manager GPLM ist neben der Ermittlung der Lastzustände GPLS auch für die Bestimmung der vermittlungstechnischen Überlastabwehrstufen OPL während der Auslastphase -overload- des Gruppenprozessors GPn zuständig. Die vermittlungstechnischen Überlaststufen OPLn sind ein Maß für die Intensität der Überlastabwehr, die während sich der Gruppenprozessor GPn in Überlast befindet, von den vermittlungstechnischen Prozessoren und der dazugehörigen Peripherie durchgeführt werden können. Die vermittlungstechnischen Überlaststufen OPL werden dabei vorzugsweise in Stufen zwischen 0 und 6 unterteilt:

- OPL 0 bedeutet keine Abwehr, bei
- OPL 1 es werden ca. 16 Prozent aller neu ankommenden Calls abgewiesen, bei
- OPL 6 werden nur noch Calls akzeptiert die über priorisierte Verbindungsleitungen von einem anderen Kommunikationssystem eintreffen.

Die hier aufgeführten Lastzustände GPLS des Gruppenprozessors GPn sowie die vermittlungstechnischen Überlaststufen OPLn können beliebig, entsprechend der Dimensionierung des Kommunikationssystems K, verfeinert oder abgerundet werden.

Ist ein Gruppenprozessor GPi nicht im Lastzustand "overload", wird kein Call weiterverteilt. Ist ein Gruppenprozessor jedoch in dem Lastzustand "overload" so wird ein Anteil der neu ankommenden Calls anhand eines Wertes q, der zwischen 0 und 1 schwankt, bestimmt der zur weiteren Abarbeitung an freie Gruppenprozessoren GPx weitergeleitet werden sollte. Durch einen weiteren ermittelbaren Wert p, der die Aufnahmefähigkeit für weitere Calls der übrigen Gruppenprozessoren wiedergibt, wird festgelegt, ob aufgrund der Gesamtauslastung dieser Gruppenprozessoren GPn des Kommunikationssystems K überhaupt verteilt werden kann. Wenn entschieden ist, daß ein Call verteilt werden soll, wird entsprechend der ermittelten Werte p der Gruppenprozessoren bestimmt, welche "Ziel Gruppenprozessoren" GPx zur Abarbeitung der ECPS Prozeduren bereitgestellt werden.

Die Werte q und p werden von jedem Gruppenprozessor GPn aus den Lastzuständen GPLS und seines eigenen Lastwertes a lokal berechnet. Nachfolgend ist ein Pseudocode zur Berechnung der Werte p und q angegeben.

```

PROCEDURE calc q;
! calculate q;
BEGIN
    IF q=0
    THEN q:=min (1, freeGPs/overloaded GPs-0.1));

```

```

IF myOLC.GPLS<2
THEN BEGIN ! GP not overloaded;
  IF q<freeGPs/(overloaded:GPs+0.01)      ! q too small;
  THEN q:=MIN(1,free GPs/(overloaded GPs+0.01))
  ELSE q:=MIN(q*p sum,1); !no distr. ness., slowly
increase q
  END
ELSE BEGIN !GP is overloaded;
  IF p sum > 1      !too much distributed;
  THEN BEGIN
    q:=q*p sum;
    damping:=damping+1;
  END
  ELSE BEGIN !more distribution possible;
    IF (myOLC.OfferedLoad>0,95 OR myOLC.OPL>0) AND
damping>2
    THEN BEGIN
      q:=MIN(q*1.2,1);
      damping:=0;
    END
    ELSE damping:=damping+1;
    IF myOLC.offeredLoad<0.65 AND myOLC.OPL=0
    THEN BEGIN
      q:=MAX(0,q/1.1);
      damping:=damping+1;
    END;
    END;
  END GP is overloaded;
END PROCEDURE calc q;
END CLASS smartGPL TAB;

PROCEDURE calc p;
! calculate p(i);
BEGIN
  INTEGER i;
  REAL alpha; !Initializationprob. for p;

```

```

alpha:=1/(4*(freeGPs+0.1));
p_sum:=0;
FOR i:=1 STEP 1 UNTIL NumberOfGPs DO BEGIN
5   IF the GPL TAB(i)=0
      THEN BEGIN
          IF p(i)=0 THEN p(i):=alpha ELSE p(i):=p(i)*1.05;
10  END;
      IF the GPL TAB(i)=1
          THEN p(i):=max(0,p(i)*0.9);
      IF the GPL TAB(i)>1
15  THEN p(i):=0;
      p_sum:=p_sum+p(i);
      END;
20  IF p_sum>0 THEN BEGIN !normalize p();
      FOR j:=1 STEP 1 UNTIL NumberOfGPs DO
          p(j):=p(j)/p_sum;
25  END
END PROCEDURE calc p;

```

In Fig. 4 ist ein Ablaufdiagramm, in dem die einzelnen Programmprozeduren angedeutet sind, die zur Abarbeitung von beispielsweise lokal auftretenden Überlasten an einen Gruppenprozessor GPj aufgerufen und abgearbeitet werden.

Die Berechnung von den im Verteilschema Fig. 3 angegebenen Werten $p_i(n)$, q_j sowie der Lastzustand $GPLS_j$ geschieht wie oben bereits erwähnt im Programmmodul Load Manager GPLM des Gruppenprozessors GPj. Der GPLM ist ein Prozeß der von der virtuellen Prozessoreinheit VCPU-OAMH abgearbeitet wird.

35 Nachfolgend werden die im Ablaufdiagramm aufgezeigten Schritte beschrieben:

- Berechnung des Lastwertes a:

Der Lastwert a wird aus der Belastung der zentralen Prozessoreinheit des jeweiligen Gruppenprozessors ermittelt.

- Bestimmung des Load States GPLS eines Gruppenprozessors GPs:

40 Zur Bestimmung des Load State GPLS wird ein integrierender Startindikator S, der in der Art eines "Schätzers" eine andauernde Überlastsituation erkennt, ein Load Balancing Flag LBF sowie x eine Hysteresebreite, der beispielsweise der Wert 0,1 zugewiesen werden kann, verwendet.

Die Bestimmung der quantifizierten Größen GPLS wird nach folgendem Pseudocode durchgeführt:

a) Bisheriger GPLS "Normal"

```

45   if a > Normallastschwelle + x
      then GP LS := "high";
      if S > Thresh and a > Hochlastschwelle + x
50  then GPLS := "overload";

```

b) Bisheriger GPLS "high"

55

```

        if a < Normallastschwelle
        then GPLS := "normal"
5         if S > Thresh und a > Hochlastschwelle + x
        then GPLS := "overload";

c)Bisheriger GPLS "overload"
10
        if LBF= true
        then a' := a/(1- q(j)d)
        else a' := a;
15         if a' > Hochlastschwelle or OPL > 0
        then begin
                GPLS:="overlast";
20                 real overlast:= true;
                if (OPL = 6 since 3s)
                Then GPLS := "extrem"
25         end
        esle begin (Ab hier liegt keine echte Überlast mehr vor)

30
                if no rejection since 6s
                then beginn
                        if a' < Hochlastschwelle
35                        then begin
                                GPLS := "high"
                                if a' < normallastschwelle
40                                then GPLS := "normal"
                                end;
                        end
                else begin
45                        GPLS := "overload";
                        real overload:= false;
                        end;
50                end;

```

d) Bisheriger GPLS = "extrem"

Von den eigenen Anschlußeinheiten LTGs wird kein neuer Lastverkehr mehr angenommen.

55 Die Programmprozedur "Bestimmung GPLS" wird im wesentlichen aus dem Lastwert a des Prozessors der letzten Sekunde bestimmt. Falls bereits verteilt wird, wird über die Größe a', die die tatsächliche an den Gruppenprozessor anliegende Last ausdrückt, berechnet. Der Lastwert a' stellt einen Schätzer für die zu erwartende Last für die zentrale Prozessoreinheit CPU des nächsten Kontrollintervalls dar.

- Informieren der Load Manager auf allen GPn über die Lastzustände aller GPn

Zwei Möglichkeiten der Informationsverteilung kommen dazu in Betracht:

a) Alle GPLMs senden ihre lokale Lastinformation (GPLS, OPL) an einen zentralisierten Prozeß, Load Control Master (LOCM) genannt, auf einem ganz bestimmten Prozessor GPx. Der LOCM sammelt die Lastinformationen und sendet so rasch als möglich die gesammelten Informationen an alle GPLMs im System.

b) Mittels einer 'jeder-informiert-jeden' Kommunikation zwischen den GPLMs verteilt jeder GPLM seine lokale Lastinformation (GPLS, OPL) an die GPLM aller anderen GPn im System.

Nach einem kurzen Zeitintervall der Zwischenspeicherung aller GPLS werden diese an alle GPn verteilt und dort jeweils in eine Tabellen eingetragen. Nach einem kurzem Zeitintervall, stehen aktualisierte GPLS von jedem Gruppenprozessor GPx zur Eintragung in eine Gruppenprozessortabelle bereit.

- Bestimmung der zusätzlich abzuarbeitenden Aufgaben p(i):

Der Wert p(i) ist dabei der Anteil der von den "Heim- Gruppenprozessoren " wegverteilten Calls C, die auf einen freien Gruppenprozessor GP(j) bearbeitet werden sollen. Der Wert p(i) wird dabei nach

folgender Strategie bestimmt:

a) GPLS(i) = "normal"

p(i) kann erhöht werden, da auf GP(i) noch Bearbeitungskapazität ausgenutzt werden kann.

b) GPLS(i) = "high"

p(i) wird verringert, da GP(i) zuviel Last hat.

c) GPLS(i) = "overload"

p(i) wird auf Null gesetzt, da GP(i) in einem Betriebszustand der Überlast ist.

Bei der Neueinstellung der Werte p(i) wird von einem durchschnittlichen Verteilanteil pro freien Gruppenprozessor GP(i) ausgegangen. Solange sich ein GP(i) im Lastzustand "overload" befindet, ist p(i) = 0. Wechselt der Lastzustand von "overload" in "normal" oder "high" so wird der Wert p(i) = 1/(C*(Anzahl der freien GP(x))) festgelegt. C ist ein Parameter, in dem vermittlungsspezifische Überlegungen betreffend der Vermeidung eines Überschwingens des Lastverteilsystems in der Einschwingphase eingeht.

- Bestimmung der wegzuverteilenden Aufgaben q(j)

Bevor der Wert q(j), der den Anteil der wegzuverteilenden Calls angibt, wenn ein Gruppenprozessor im Lastzustand Überlast ist, wird an Hand der p(i) festgestellt ob überhaupt verteilt werden kann.

Ergibt die Summe der noch nicht normierten p(n) einen Wert der größer als 1 ist, so wird der Wert q(j) nach den Erfordernissen des jeweiligen GP(j) bestimmt. Anhand des Lastwertes a' wird festgestellt, ob q(j) erhöht oder erniedrigt werden sollte. Falls der GP(j) nicht im Lastzustand "overload" ist, wird immer versucht q(j) zu erhöhen.

a) Falls die vermittlungstechnische Überlaststufe OPL > 0 oder Lastwert a > 0,8 + x (mit x = 0,1) ist, wird q(j) erhöht.

b) Falls GPLS(j) = "overload" und OPL = 0 und Lastwert a < 0,8 - x ist wird q(j) verringert.

c)

Falls GPLS(i) < "overload" erhöhe q(i)

Die Bestimmung des Wertes q(j) im Fall c bezieht sich auf den Einstieg in die Lastverteilung.

Normalerweise sollte dann q(j) = 1 sein, um bei plötzlicher Überlast maximale Verteilung zu bekommen. Wenn das Kommunikationssystem K an seiner Leistungsgrenze arbeitet, muß q(j) kleiner sein, um die Lastverteilung, durch Eintritt zusätzlicher Gruppenprozessoren GPx in Überlast nicht zu gefährden. Aus diesem Grund wird q(j) verringert, falls die Summe der einzelnen p(n) kleiner als 1 ist.

Um mögliche Schwingungen des Überlastverfahrens zu vermeiden, ist es vorteilhaft, wenn vor einer Verringerung des Werte q(j) ein kurzer Zeitintervall abgewartet wird. Nachdem festgestellt wurde, daß der Gruppenprozessor GP(j) Calls verteilen möchte, wird das Load Balancing Flag LBF "true", wenn nicht, wird das LBF "false" gesetzt. In einem mit Guard Period bezeichneten Schutzintervall, wird das Load Balancing Flag LBF := false gesetzt, obgleich der Gruppenprozessor GP(j) sich im Lastzustand "overload" befinden kann.

Bei einer Neueinstellung des Wertes q(j) wird folgender Algorithmus angewendet:

Falls q(j) = 0 wird q(j) := min (1, (Anzahl freier GPx/ Anzahl der GPs die im Lastzustand "overload" sind)).

Ist der GPLS von GP(j) nicht im "overload" und q(j) < (Anzahl freier GPs /Anzahl der GPs in "overload"), so wird q(j) := min(1, Anzahl freier GPs/Anzahl GPs in "overload") gesetzt. Wenn der GP(j) nicht in "overload" ist, ist normalerweise q(j) = 1, um einen schnellen Einstieg in die Lastverteilung zu gewährleisten, falls der GP(j) plötzlich in Überlast gerät. Falls jedoch mehr GPs in "overload" sind als GPs zur Aufnahme von

fremder Überlast bereitstehen, wird $q(j)$ passend kleiner gewählt, um die nicht überlasteten GPs vor Überlast zu schützen und damit das System der Lastverteilung nicht zu gefährden.

Das Verteilen der Calls erfolgt wie nachfolgend angegeben: Beim Anliegen eines Calls, stellt der Service Manager SM, der eine direkte Schnittstelle zu den zu bearbeitenden Calls darstellt und die eigentliche Verteilung der Calls vornimmt anhand des LBF fest, ob verteilt werden soll oder nicht. Falls verteilt werden soll, wird untersucht, ob es möglich ist den anliegen Call zu verteilen. Falls die Verteilung im Prinzip möglich und erwünscht ist, wird entschieden, ob dieser spezielle Call verteilt werden soll. Dazu wird der Quotient aus den bisher in dem Zeitintervall verteilten Calls mit der Anzahl aller ankommenden Calls gebildet und mit $q(j)$ verglichen. Ist der Quotient größer als $q(j)$, wird der Call verteilt. Ist beispielsweise die Anzahl der bisher angekommenen Calls 20, die der bisher verteilten Calls 4 und der Wert $q(i)=0,25$, so ergibt sich aus dem Quotienten $4/20$, daß dieser kleiner als $q(j)$ ist. In diesem Fall wird der Call verteilt. Wenn der Call verteilt werden soll, wird dazu analog anhand der Werte von $p(j)$ entschieden, welcher Gruppenprozessor GPx die Bearbeitung des Calls übernimmt.

15 Patentansprüche

1. Verfahren zum Lastenausgleich in einem Multiprozessorsystem, insbesondere ein Multiprozessorsystem eines Kommunikationssystems (K), bei dem anfallende Aufgaben von mehreren Prozessoren (GPn) unter Realzeitbedingungen abgearbeitet werden können,
mit folgenden Verfahrensschritten:
 - jeder Prozessor(GPi) ermittelt seinen Lastzustand(GPLSi) in Form einer quantifizierten Größe,
 - jedem Prozessor(GPi) werden die Lastzustände(GPLSn) der anderen Prozessoren(GPn) innerhalb eines Zeitrasters mitgeteilt,
 - jeder Prozessor(GPi) gibt in Abhängigkeit vom Überschreiten einer bestimmten Größe seines Lastzustandes(GPLSi) und in Abhängigkeit von den Lastzuständen der übrigen Prozessoren(GPn) zumindest einen Teil der bei ihm anfallenden Aufgaben an die übrigen Prozessoren(GPn) ab,
 - die abgegebenen Aufgaben werden entsprechend der Lastzustände (GPLSn) der übrigen Prozessoren(GPn) auf diese aufgeteilt.
2. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß ein Prozessor(GPi) nur so viele Aufgaben abgibt, daß sein Lastzustand(GPLS) wieder die genannte bestimmte Größe unterschreitet.
3. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß die die Lastzustände charakterisierenden quantifizierten Größen jeweils in deren oberen sowie unteren Bereichsübergängen mit einem Hysteresewert versehen werden.
4. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß bei Abgabe von Aufgaben von einem Prozessor(GPi) an die übrigen Prozessoren (GPn) zusätzlich ein Maß für eine Aufgabenabwehr des Prozessors(GPi) bestimmt wird.
5. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß die prozessorbezogenen Lastzustände (GPLSn) mittels einer 'jeder-informiert-jeden' Kommunikation von den einzelnen Control Managern der Prozessoren (GPn) direkt an alle anderen Control Managern der Prozessoren (GPn) im Kommunikationssystem (K) verteilt werden.
6. Verfahren nach Anspruch 1,
dadurch gekennzeichnet,
daß die prozessorbezogenen Lastzustände(GPLSn) im Load Control Manager (LOCM) eines bestimmten Prozessors(GPx) zusammengefaßt werden.
7. Verfahren nach Anspruch 6,
dadurch gekennzeichnet,

daß entsprechend eines Zeitrasters die in dem bestimmten Prozessor(GPx) zusammengefaßten Lastzustände(GPLSn) jeweils in Tabellen der Prozessoren(GPn) abgespeichert werden.

- 5 8. Verfahren nach einem der vorhergehenden Ansprüche,
dadurch gekennzeichnet,
daß in Abhängigkeit vom jeweiligen Lastzustand (GPLS) der Prozessoren der Anteil ($p_i(n)$) der zusätzlich durch sie noch abarbeitbaren Aufgaben festgelegt wird.
- 10 9. Verfahren nach einem der vorherigen Ansprüche,
dadurch gekennzeichnet,
daß in Abhängigkeit vom jeweiligen Lastzustand (GPLSn) der Prozessoren (GPi) die von diesem wegzuverteilenden Aufgaben ($q(j)$) erhöht oder vermindert werden.
- 15 10. Verfahren nach einem der vorherigen Ansprüche,
dadurch gekennzeichnet,
daß in jedem Prozessor in Abhängigkeit vom Summenergebnis, aus den Anteilen ($p(i)$) der durch die übrigen Prozessoren zusätzlich zu übernehmenden Aufgaben, der Anteil der wegzuverteilenden Aufgaben ($q(j)$) im Überlastfall bestimmt wird.

20

25

30

35

40

45

50

55

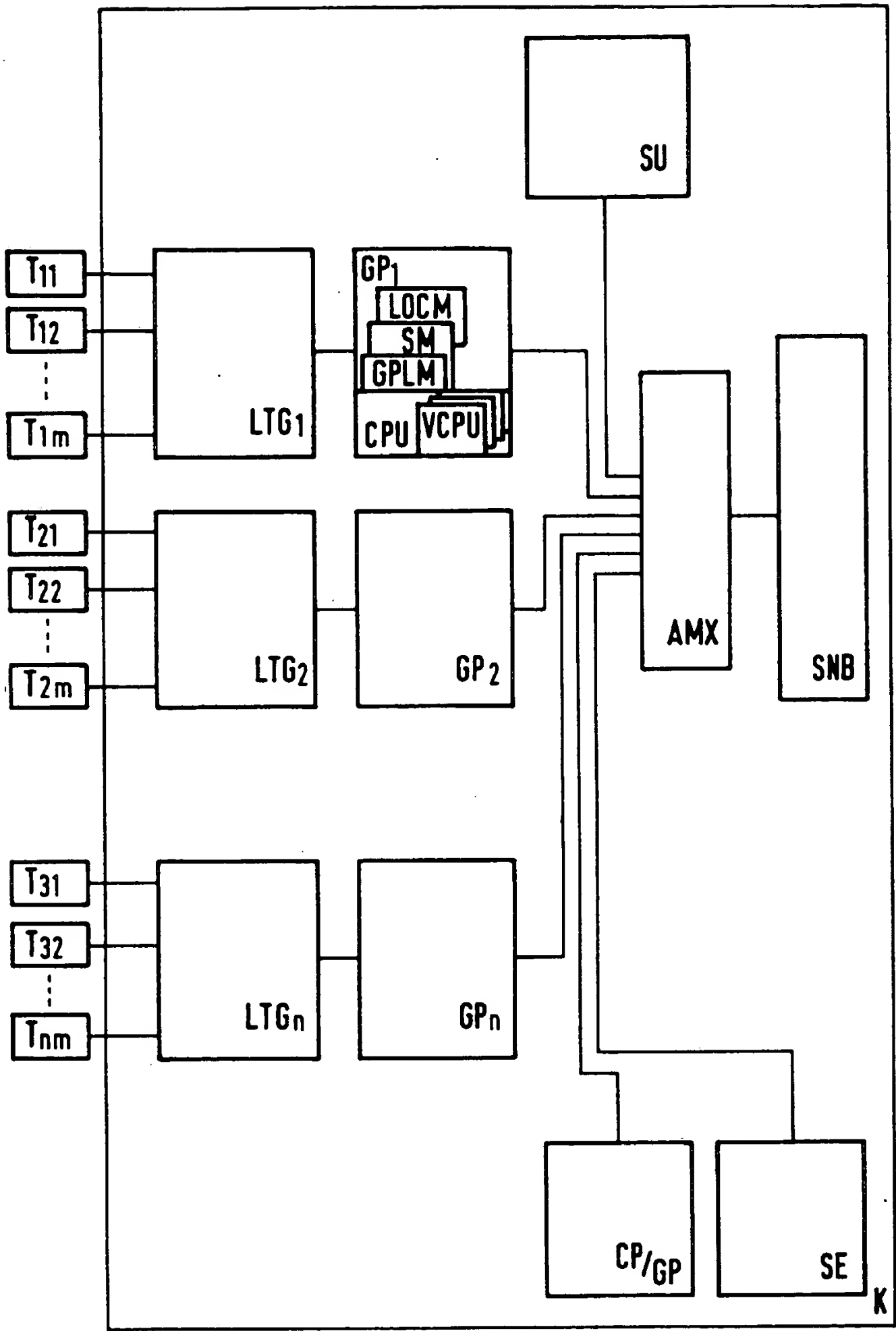


FIG 1

FIG 2

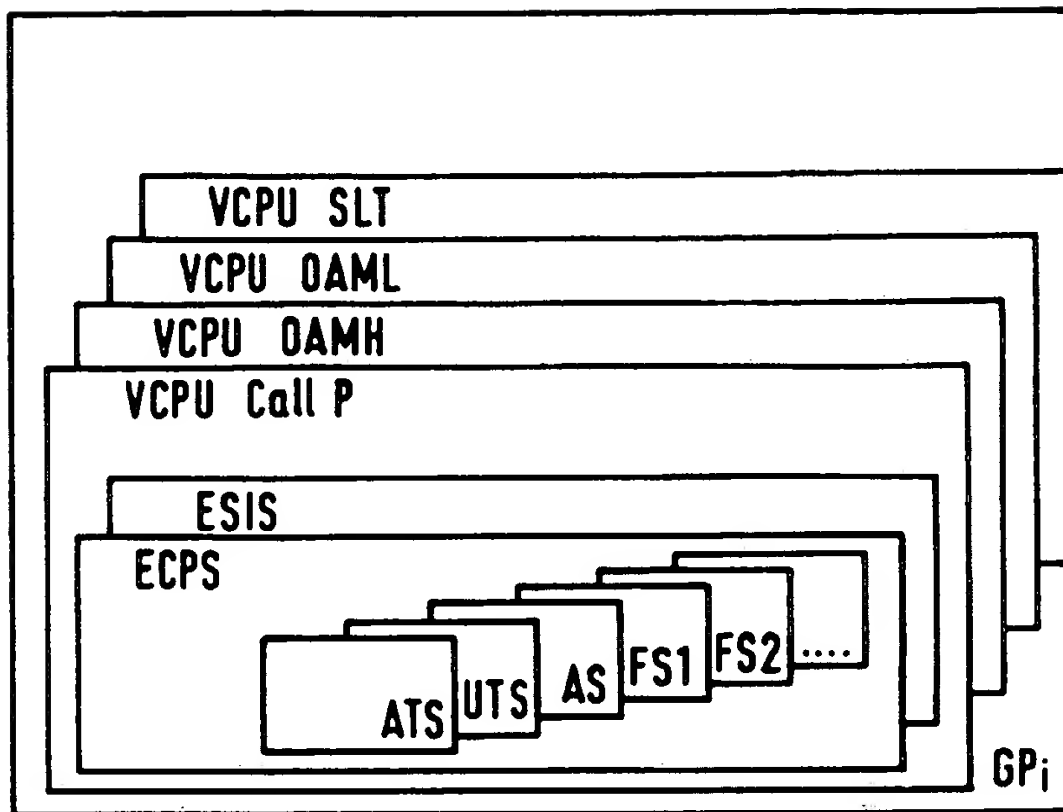


FIG 3

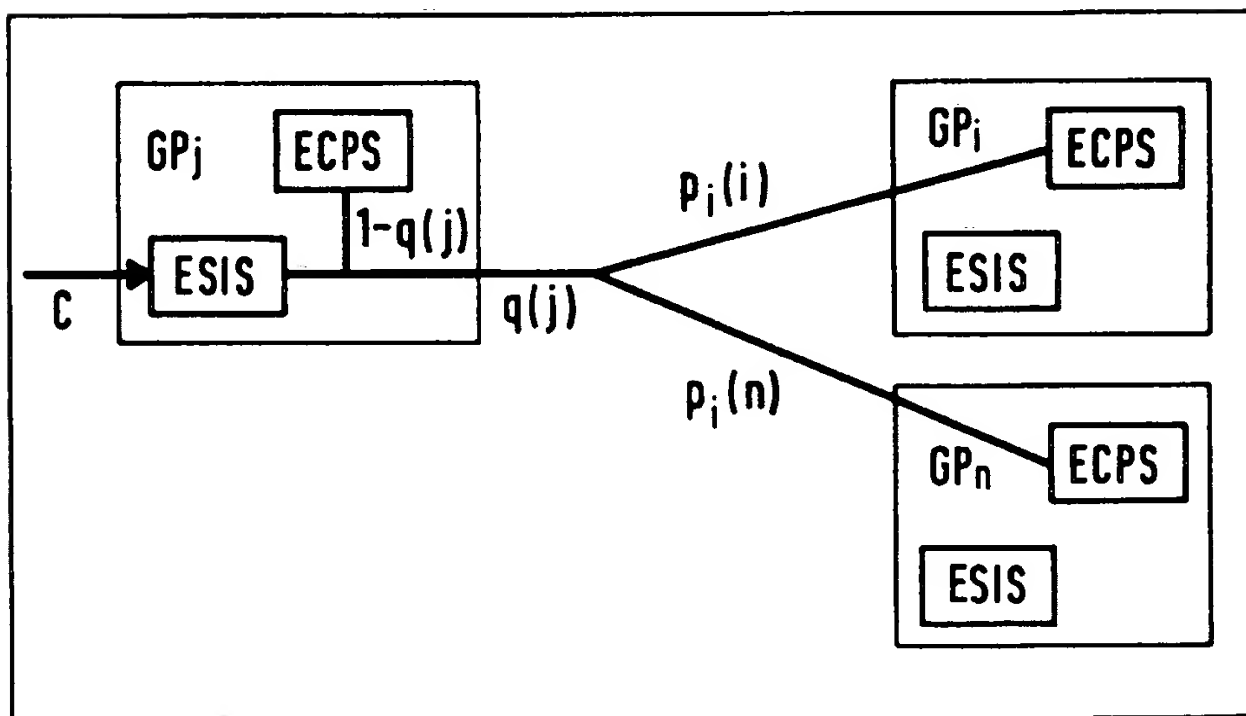
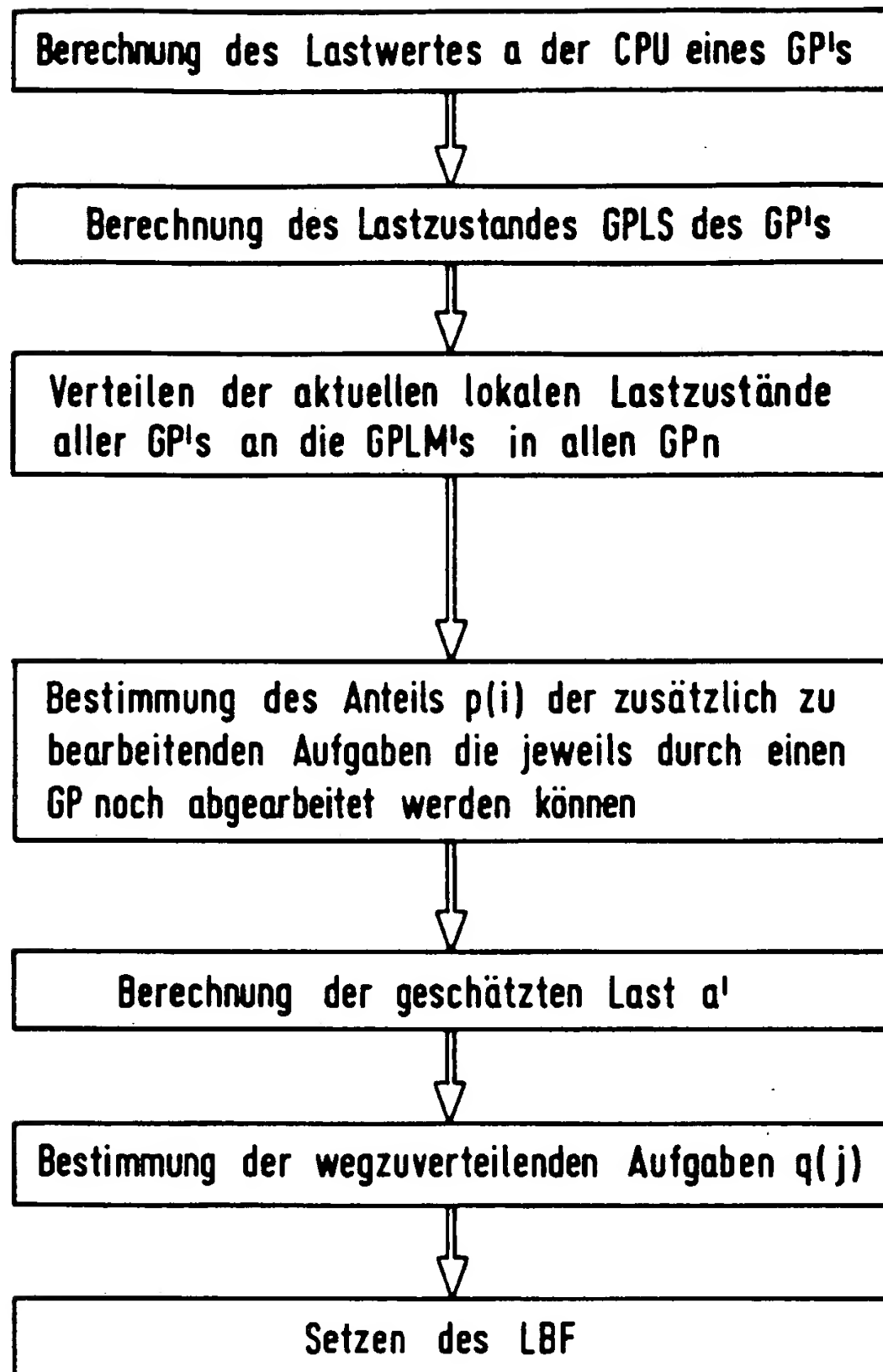


FIG 4





Europäisches
Patentamt

EUROPÄISCHER RECHERCHENBERICHT

Nummer der Anmeldung
EP 93 11 5467

EINSCHLÄGIGE DOKUMENTE			
Kategorie	Kennzeichnung des Dokuments mit Angabe, soweit erforderlich, der maßgeblichen Teile	Betrifft Anspruch	KLASSIFIKATION DER ANMELDUNG (Int.Cl.6)
X	SOFTWARE PRACTICE & EXPERIENCE Bd. 19, Nr. 5, Mai 1989, CHICHESTER GB Seiten 411 - 435 XP98633 PIYUSH DIKSHIT ET AL. 'SAHAYOG: a test bed for evaluating dynamic load-sharing policies' * Seite 419, Zeile 6 - Zeile 19 * * Seite 425, Zeile 21 - Seite 426, Zeile 10 *	1,2,5-7, 9	G06F9/46 H04M3/36 H04Q3/545
Y	---	3,4,8,10	
Y	IEEE TRANSACTIONS ON COMPUTERS Bd. 38, Nr. 8, August 1989, NEW YORK US Seiten 1124 - 1142 XP47581 KANG G. SHIN ET AL. 'Load sharing in distributed real-time systems with state-change broadcasts' * Seite 1124, linke Spalte, Zeile 1 - Seite 1125, rechte Spalte, Zeile 10 * * Seite 1125, rechte Spalte, Zeile 32 - Seite 1126, linke Spalte, Zeile 15 * * Seite 1127, linke Spalte, Zeile 16 - Zeile 58 *	3,4,8,10	
	---		RECHERCHIERTE SACHGEBIETE (Int.Cl.6)
A	PROC. 12TH INT. CONF. ON DISTRIBUTED COMPUTING SYSTEMS 9. Juni 1992, IEEE, LOS ALAMITOS, US Seiten 82 - 89 XP341000 JUNGUK KIM ET AL. 'An all-sharing load balancing protocol in distributed systems on the CSMA/CD local area network' * Seite 83, linke Spalte, Zeile 52 - rechte Spalte, Zeile 16 * * Seite 84, linke Spalte, Zeile 35 - rechte Spalte, Zeile 29 * --- -/--	10	G06F H04M H04Q
Der vorliegende Recherchenbericht wurde für alle Patentansprüche erstellt			
Recherchenort DEN HAAG		Abschlußdatum der Recherche 26. November 1993	Prüfer Kingma, Y
KATEGORIE DER GENANNTEN DOKUMENTE X : von besonderer Bedeutung allein betrachtet Y : von besonderer Bedeutung in Verbindung mit einer anderen Veröffentlichung derselben Kategorie A : technologischer Hintergrund O : mündliche Offenbarung P : Zwischenliteratur T : der Erfindung zugrunde liegende Theorien oder Grundsätze E : älteres Patentdokument, das jedoch erst am oder nach dem Anmeldedatum veröffentlicht worden ist D : in der Anmeldung angeführtes Dokument I : aus andern Gründen angeführtes Dokument & : Mitglied der gleichen Patentfamilie, übereinstimmendes Dokument			



Europäisches
Patentamt

EUROPÄISCHER RECHERCHENBERICHT

Nummer der Anmeldung
EP 93 11 5467

EINSCHLÄGIGE DOKUMENTE			
Kategorie	Kennzeichnung des Dokuments mit Angabe, soweit erforderlich, der maßgeblichen Teile	Betrifft Anspruch	KLASSIFIKATION DER ANMELDUNG (Int.Cl.6)
A	PROC. 11TH INT. TELETRAFFIC CONGRESS 4. September 1985 , ELSEVIER, AMSTERDAM, NETHERLANDS Seiten 835 - 841 M. VILLEN ALTAMIRANO ET AL. 'An overload control strategy for distributed control systems' * das ganze Dokument *	2,8	
A	US-A-4 974 256 (BERNARD L. CYR ET AL.) * Spalte 2, Zeile 5 - Spalte 3, Zeile 13 *	4,8-10	
Der vorliegende Recherchenbericht wurde für alle Patentansprüche erstellt			RECHERCHIERTE SACHGEBIETE (Int.Cl.6)
Recherchenort DEN HAAG		Abschlußdatum der Recherche 26. November 1993	Prüfer Kingma, Y
KATEGORIE DER GENANNTEN DOKUMENTE X : von besonderer Bedeutung allein betrachtet Y : von besonderer Bedeutung in Verbindung mit einer anderen Veröffentlichung derselben Kategorie A : technologischer Hintergrund O : mündliche Offenbarung P : Zwischenliteratur T : der Erfindung zugrunde liegende Theorien oder Grundsätze E : älteres Patentedokument, das jedoch erst am oder nach dem Anmeldedatum veröffentlicht worden ist D : in der Anmeldung angeführtes Dokument L : aus andern Gründen angeführtes Dokument & : Mitglied der gleichen Patentfamilie, übereinstimmendes Dokument			